

XFOX760 Driver Manual

Foxboro 760CNA/760CSA Single Station Micro Controllers Driver



CPKSoft Engineering Process Monitoring and Industrial Automation Software

Copyright 1990-2008, CPKSoft Engineering. All rights reserved.

Index

1.	Introduction	3
2.	Driver details	4
2.1.	Driver overview	4
2.2.	Supported devices.....	4
3.	Command list	5
3.1.	Read Current Set Point, Measurement, Output, Status and Alarm Values	5
3.2.	Read Controller Inputs #1 to #4	6
3.3.	Write Current Set Point, Measurement, Output, Status or Alarm Values.....	7
3.4.	Write Setpoint Value	8
3.5.	Write Output Current Value	9
3.6.	Set Controller Mode	9
3.7.	Set Controller in Manual Mode	10
3.8.	Set Controller in Auto Mode	10
3.9.	Alarm Acknowledge.....	11
3.10.	Write Controller Memory As Bytes	11
3.11.	Write Controller Memory As Bits	12
3.12.	Read Controller Memory As Bytes	12
3.13.	Read Controller Memory As Bits	13
3.14.	Read Controller Memory As Words.....	13
3.15.	Write Controller Memory As Words.....	14
4.	Appendices	15
4.1.	Error messages	15
4.2.	Keywords list.....	15

1. Introduction

CPKSoft Engineering assumes no responsibility for any errors that may appear in this document. If you have any suggestions for improvements or amendments or have found errors in this publication, please notify us.

This driver is included with all unlimited licenses of TAS-HMITalk. It is not sold separately. It requires the TAS-HMITalk ActiveX to work, therefore it cannot be used as a stand-alone driver.

If you use this driver in your applications, you need to include the xfox760.tlk in the set of files that you distribute. This file must be located in the same folder where the hmitalk.ocx file is registered in order to be found by the activex when the applications are executed.

The source-code for the xfox760.tlk driver is available in plain-C language for additional USD 699 if you own a license of TAS-HMITalk 8.04 or higher.

Refer to the following link to visit the xfox760 driver page at CPKSoft Engineering website: <http://www.cpksoft.com/tabid/55/ProductID/36/PageIndex/1/Default.aspx>.

Visit this link if you want to see a complete list of drivers that are currently available for TAS-HMITak: <http://www.cpksoft.com/Drivers/tabid/55/Default.aspx>.

Also, refer to this link if you are interested in purchasing a license of the most recent version of TAS-HMITalk: <http://www.cpksoft.com/Products/tabid/54/Default.aspx>.

We welcome your comments about this document. You can reach us by e-mail at [contact @ cpksoft.com](mailto:contact@cpksoft.com).

2. Driver details

2.1. Driver overview

XFOX760 driver allows you to communicate with the FOXBORO 760CNA and 760CSA SINGLE STATION MICRO Controllers. This driver has been developed according to the Foxboro Instruction Book 2848.

Cable Wiring:

Use pins 24 and 25 in the back of the controller to connect to the RS485 serial port of the controller. Check this with your controller's manual to see if this pins have not changed from previous versions of the controller. Use an RS-232/485 converter to connect the controller to your PC. Set the CommHoldRTSWhileTransmitting and CommHoldRTSWhileReceiving properties to On. Make sure that the controller station address, baudrate, parity, databits and stop bits are correctly configured in the driver.

2.2. Supported devices

This driver can communicate with these devices, but is not necessarily limited to this list:

FOXBORO 760CNA Single Station Micro Controllers
FOXBORO 760CSA Single Station Micro Controllers

3. Command list

3.1. Read Current Set Point, Measurement, Output, Status and Alarm Values

Description of this command:

Reads the current values for the set point, measurement, and output, plus additional controller status and alarm information.

Type of data handled by this command:

Analog Input

Number of points accepted by this command:

1-19

Meaning of the DriverP0 parameter:

Controller Address (0-99).

Meaning of the DriverP1 parameter:

11

Values that are returned:

Value in PointValue (0) = Primary controller set point.

Value in PointValue (1) = Primary controller measured value.

Value in PointValue (2) = Primary controller output.

Value in PointValue (3) = FLAG BYTE/Bit 0: User interface entered indicator:

0 = not entered

1 = entered subsequent to last host acknowledgment

Value in PointValue (4) = FLAG BYTE/Bit 1: Primary A/M setting

0 = MANUAL

1 = AUTO

Value in PointValue (5) = FLAG BYTE/Bit 2: W/P setting

0 = PANEL

1 = WORKSTATION

Value in PointValue (6) = FLAG BYTE/Bit 3: Primary R/L setting

0 = LOCAL

1 = REMOTE

Value in PointValue (7) = FLAG BYTE/Bit 4: Reserved

Value in PointValue (8) = FLAG BYTE/Bit 5: Reserved

Value in PointValue (9) = FLAG BYTE/Bit 6: Reserved

Value in PointValue (10) = FLAG BYTE/Bit 7: Alarm indicator

0 = No alarms

1 = Alarm present

Value in PointValue (11) = ALARM BYTE/Bit 0: (Only valid if BIT 1 = 1)

0 = OUTPUT alarm is LOW alarm
1 = OUTPUT alarm is HIGH alarm
Value in PointValue (12) = ALARM BYTE/Bit 1:
0 = No OUTPUT alarm exists
1 = OUTPUT alarm exists
Value in PointValue (13) = ALARM BYTE/Bit 2: (Only valid if BIT 3 = 1)
0 = DEVIATION alarm is LOW alarm
1 = DEVIATION alarm is HIGH alarm
Value in PointValue (14) = ALARM BYTE/Bit 3:
0 = No DEVIATION alarm exists
1 = DEVIATION alarm exists
Value in PointValue (15) = ALARM BYTE/Bit 4: (Only valid if BIT 5 = 1)
0 = MEAS 2 alarm is LOW alarm
1 = MEAS 2 alarm is HIGH alarm
Value in PointValue (16) = ALARM BYTE/Bit 5:
0 = No MEAS 2 alarm exists
1 = MEAS 2 alarm exists
Value in PointValue (17) = ALARM BYTE/Bit 6: (Only valid if BIT 7 = 1)
0 = MEAS 1 alarm is LOW alarm
1 = MEAS 1 alarm is HIGH alarm
Value in PointValue (18) = ALARM BYTE/Bit 7:
0 = No MEAS 1 alarm exists
1 = MEAS 1 alarm exists

3.2. Read Controller Inputs #1 to #4

Description of this command:

Reads analog inputs #1 to #4 current values.

Type of data handled by this command:

Analog Input

Number of points accepted by this command:

1-4

Meaning of the DriverP0 parameter:

Controller Address (0-99).

Meaning of the DriverP1 parameter:

32

Meaning of the DriverP2 parameter:

1053

Meaning of the DriverP3 parameter:

-40

3.3. Write Current Set Point, Measurement, Output, Status or Alarm Values

Description of this command:

Writes the setpoint value or the primary controller output, as well as to change the status of some flags like A/M, R/L and to acknowledge alarms in the controller.

Type of data handled by this command:

Analog Output

Number of points accepted by this command:

1

Meaning of the DriverP0 parameter:

Controller Address (0-99).

Meaning of the DriverP1 parameter:

12

Meaning of the DriverP2 parameter:

Defines the STATUS BYTE (here it acts like a sub-command). This command is used to change the setpoint value or the primary controller output, as well as to change the status of some flags like A/M, R/L and to acknowledge alarms in the controller. There is no command which allows you to change the setpoint or output only, without in turn having to indicate some other related data. Thus, it is necessary to specify in all cases the STATUS BYTE through HMITalk1.DriverP2. According to the value provided in HMITalk1.DriverP2, the current Analog Output value may or may not be used. The Analog Output value is only used and sent to the controller in those cases where the STATUS BYTE BIT0 is 1. At that point, BIT5 decides if what is to be changed is the setpoint or the output. The Analog Output value must be taken to internal controller format like in the case of the WRITE COMMAND. If BIT0 is 0, the Analog Output value is not sent and the command is used to change some operation flag or to be notified of the alarm status.

STATUS BYTE:

- Bit 0: Change Indicator

0 = No new output or setpoint

1 = New output or setpoint is being specified via bits 4, 5 & 6 if incremental, or value if absolute.

- Bit 1: A/M Setting:

0 = MANUAL

1 = AUTO

- Bit 2: User interface acknowledgement:

0 = No acknowledgement.

1 = Acknowledgement.

- Bit 3: R/L Setting: (R/L is ignored if the controller is configured for LOCAL only).

0 = LOCAL

1 = REMOTE

- Bit 4: Size of step change:

0 = SMALL STEP

- 1 = LARGE STEP
 - Bit 5: Output Vs setpoint:
- 0 = Change output
- 1 = Change setpoint
 - Bit 6: Direction of change:
- 0 = Increment the setting.
- 1 = Decrement the setting.
 - Bit 7: Alarm acknowledge:
- 0 = No acknowledge.
- 1 = acknowledge all current alarms.

Examples:

Meaning of the DriverP2 parameter:

01h ==> Set MANUAL, LOCAL, WORKSTATION modes and without alarm acknowledgement, sending the Analog Output value as the new controller output. (Appropriate scaling is 0, 1, 0, 40)

Meaning of the DriverP2 parameter:

20h ==> Set MANUAL, LOCAL, WORKSTATION modes and without alarm acknowledgement.

Meaning of the DriverP2 parameter:

21h ==> Set MANUAL, LOCAL, WORKSTATION modes and without alarm acknowledgement, sending the Analog Output value as the new setpoint. (Appropriate scaling is 0, 1, 0, 40)

Meaning of the DriverP2 parameter:

22h ==> Set AUTOMATIC, LOCAL, WORKSTATION modes and without alarm acknowledgement.

Meaning of the DriverP2 parameter:

23h ==> Set AUTOMATIC, LOCAL, WORKSTATION modes and without alarm acknowledgement, but sending the Analog Output value as the new setpoint. (Appropriate scaling is 0, 1, 0, 40)

Meaning of the DriverP2 parameter:

A2h ==> Set AUTOMATIC, LOCAL, WORKSTATION mode and with alarm acknowledgement. (Note that in the previous commands, 80h is added to any previous command for alarm acknowledgement).

Important note: R/L is ignored if the controller is configured for LOCAL only.

3.4. Write Setpoint Value

Description of this command:

Writes the setpoint value in the controller.

Type of data handled by this command:

Analog Output

Number of points accepted by this command:

1

Meaning of the DriverP0 parameter:

Controller Address (0-99).

Meaning of the DriverP1 parameter:

12

Meaning of the DriverP2 parameter:

35

3.5. Write Output Current Value

Description of this command:

Writes the output current value in the controller.

Type of data handled by this command:

Analog Output

Number of points accepted by this command:

1

Meaning of the DriverP0 parameter:

Controller Address (0-99).

Meaning of the DriverP1 parameter:

12

Meaning of the DriverP2 parameter:

1

3.6. Set Controller Mode

Description of this command:

Sets the controller in manual or auto mode.

Type of data handled by this command:

Digital Output

Number of points accepted by this command:

1

Meaning of the DriverP0 parameter:

Controller Address (0-99).

Meaning of the DriverP1 parameter:

12

Values that are sent:

Value in PointValue (0) = 0 sets manual mode

Value in PointValue (0) = 1 sets auto mode

3.7. Set Controller in Manual Mode

Description of this command:

Sets the controller in manual mode.

Type of data handled by this command:

Analog Output

Number of points accepted by this command:

1

Meaning of the DriverP0 parameter:

Controller Address (0-99).

Meaning of the DriverP1 parameter:

12

Meaning of the DriverP2 parameter:

32

3.8. Set Controller in Auto Mode

Description of this command:

Sets the controller in auto mode.

Type of data handled by this command:

Analog Output

Number of points accepted by this command:

1

Meaning of the DriverP0 parameter:

Controller Address (0-99).

Meaning of the DriverP1 parameter:

12

Meaning of the DriverP2 parameter:

34

3.9. Alarm Acknowledge

Description of this command:

Writes the setpoint value in the controller.

Type of data handled by this command:

Analog Output

Number of points accepted by this command:

1

Meaning of the DriverP0 parameter:

Controller Address (0-99).

Meaning of the DriverP1 parameter:

12

Meaning of the DriverP2 parameter:

35

3.10. Write Controller Memory As Bytes

Description of this command:

This command allows you to write bytes of the controller memory. The controller's memory map information is included in Appendix A of the Controller Manual.

Type of data handled by this command:

Analog Output

Number of points accepted by this command:

1-125

Meaning of the DriverP0 parameter:

Controller Address (0-99).

Meaning of the DriverP1 parameter:

13

Meaning of the DriverP2 parameter:

Initial memory address pointed. (See Appendix A)

Meaning of the DriverP3 parameter:

Conversion Factor. (See Appendix A)

3.11. Write Controller Memory As Bits

Description of this command:

Writes the controller memory byte as bits. The controller's memory map information is included in Appendix A of the Controller Manual.

Type of data handled by this command:

Digital Output

Number of points accepted by this command:

1-8

Meaning of the DriverP0 parameter:

Controller Address (0-99).

Meaning of the DriverP1 parameter:

13

Meaning of the DriverP2 parameter:

Memory address pointed. (See Appendix A)

3.12. Read Controller Memory As Bytes

Description of this command:

This command allows you to read bytes from the controller memory. The controller's memory map information is included in Appendix A of the Controller Manual.

Type of data handled by this command:

Analog Input

Number of points accepted by this command:

1-250

Meaning of the DriverP0 parameter:

Controller Address (0-99).

Meaning of the DriverP1 parameter:

14

Meaning of the DriverP2 parameter:

Initial memory address pointed. (See Appendix A)

Meaning of the DriverP3 parameter:

Conversion Factor. (See Appendix A)

3.13. Read Controller Memory As Bits

Description of this command:

This command allows you to read bytes from the controller memory as bits. The controller's memory map information is included in Appendix A of the Controller Manual.

Type of data handled by this command:

Digital Input

Number of points accepted by this command:

1-250

Meaning of the DriverP0 parameter:

Controller Address (0-99).

Meaning of the DriverP1 parameter:

14

Meaning of the DriverP2 parameter:

Initial memory address pointed. (See Appendix A)

3.14. Read Controller Memory As Words

Description of this command:

Reads consecutive words from the controller memory. It is a non-standard function which is very useful to read parameters whose values are formed by 2 bytes. In the case of packages, be careful to note that all the data required are word-type, otherwise wrong results will be obtained. The controller's memory map information is included in Appendix A of the Controller Manual.

Type of data handled by this command:

Analog Input

Number of points accepted by this command:

1-125

Meaning of the DriverP0 parameter:

Controller Address (0-99).

Meaning of the DriverP1 parameter:

32

Meaning of the DriverP2 parameter:

Initial memory address pointed. (See Appendix A)

Meaning of the DriverP3 parameter:

Conversion Factor. (See Appendix A)

3.15. Write Controller Memory As Words

Description of this command:

Writes consecutive words to the controller memory. It is a non-standard function which is very useful to write parameters whose values are formed by 2 bytes. In the case of packages, be careful to note that all destination parameters are word-type, otherwise wrong results will be obtained. The controller's memory map information is included in Appendix A of the Controller Manual.

Type of data handled by this command:

Analog Output

Number of points accepted by this command:

1-125

Meaning of the DriverP0 parameter:

Controller Address (0-99).

Meaning of the DriverP1 parameter:

33

Meaning of the DriverP2 parameter:

Initial memory address pointed. (See Appendix A)

Meaning of the DriverP3 parameter:

Conversion Factor. (See Appendix A)

4. Appendices

4.1. Error messages

The following list shows all the possible error messages that can be returned by the protocol driver during a failed communication in the 'DriverStatus' property.

This list does not include some error messages that can be returned by the activex component while attempting to establish a connection.

- [1005] DRIVER (Internal): Invalid driver stage
- [1300] PROTOCOL (Timeout): No answer
- [1410] PROTOCOL (Format): Invalid device id in response
- [2147] CONFIG (NumValues): Only one value can be read or written
- [2181] CONFIG (NumValues): Too many values (max=12)
- [2185] CONFIG (NumValues): Too many values (max=125)
- [2193] CONFIG (NumValues): Too many values (max=19)
- [2203] CONFIG (NumValues): Too many values (max=250)
- [2235] CONFIG (NumValues): Too many values (max=8)
- [3018] CONFIG (P0): Invalid device address (0-99)
- [3508] CONFIG (P1): Invalid command
- [8073] CONFIG (Remote): Controller is in PANEL mode
- [8183] CONFIG (Remote): Index requested too small
- [8191] CONFIG (Remote): Invalid command byte
- [8194] CONFIG (Remote): Invalid data
- [8220] CONFIG (Remote): No data given with command
- [8225] CONFIG (Remote): No permission for download
- [8339] CONFIG (Remote): Transmission error
- [8367] CONFIG (Remote): Wrong number of data bytes

4.2. Keywords list

The following list shows a set of words directly related to this driver.

"760CNA, 760CSA, Controllers, FOXBORO, Micro, Single, Station".